In-SRAM Boolean Operations

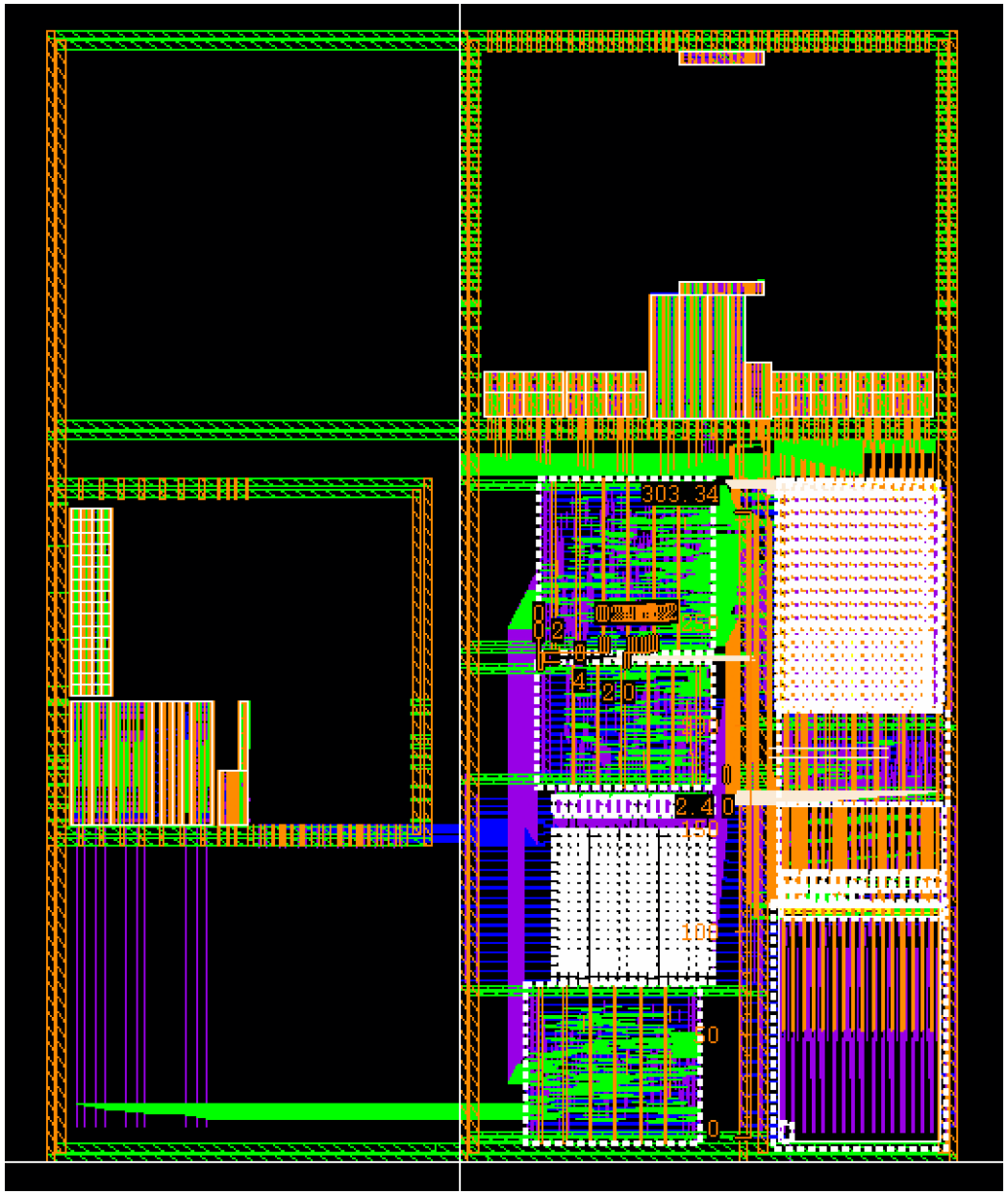
Group 3

Yu-Sheng Ting, Cheng-Lin Lee, Xinxin Wang, I-Kang Wu, Yufan Gao, Ganrun Xu

ECE ELECTRICAL & COMPUTER ENGINEERING
UNIVERSITY OF MICHIGAN

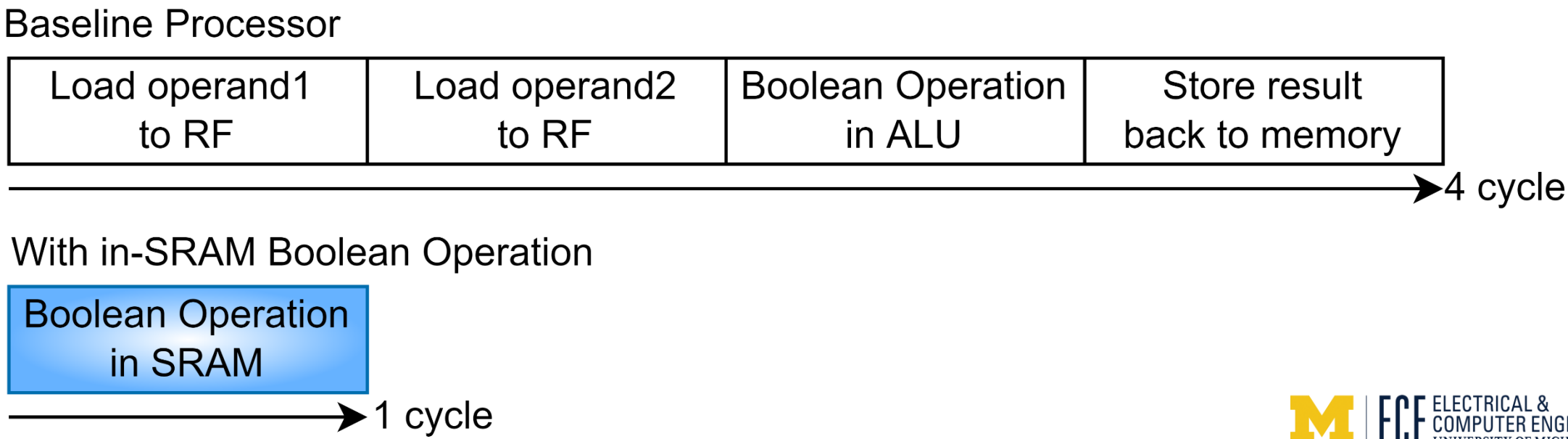
Outline

- Introduction
- Design and Implementation
  - In-SRAM Operation
  - Cell Size Specification
  - Data Disturbing
  - Integration with Baseline Processor
- Simulation Result
  - SRAM functionality
  - SRAM with the Baseline Processor
  - Static Noise Margin Analysis
- Conclusion



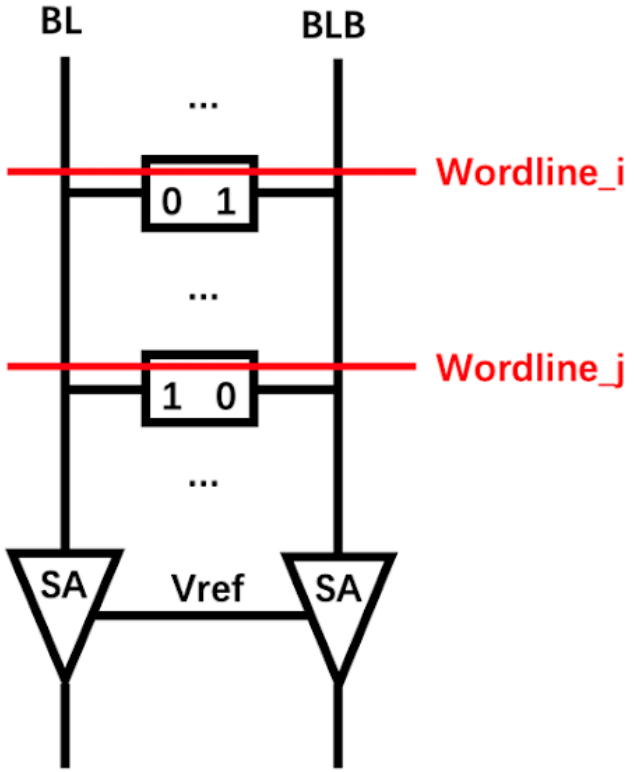
Introduction

- Motivation: Moving data over hierachy takes time in baseline processor
- In-SRAM boolean operations help to reduce the total execution time by avoiding the data movement



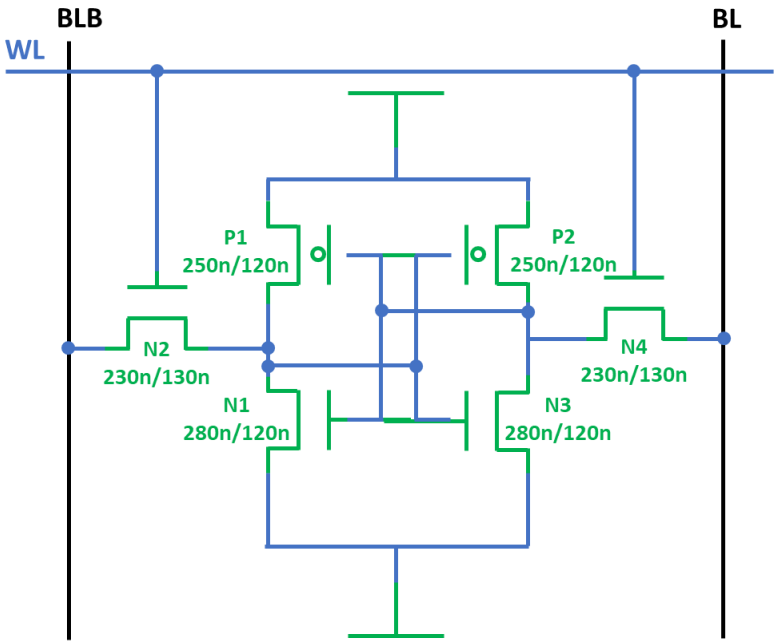
In-SRAM Operation

- Bit-line computation technology
  - AND → BL
  - NOR → BLB
- AND/NOR
  - BL/BLB pre-charge to high
  - Active two wordlines
  - Sense the change on BL/BLB



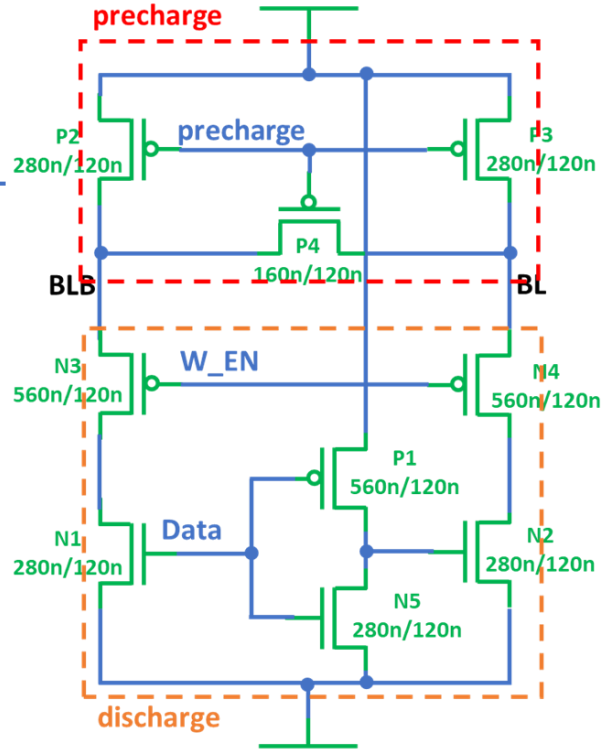
Cell Size Specification

- Read stability: N1/N3>N2/N4
- Leakage control: N2/N4 length↑
- Write stability: N2/N4>P1/P2



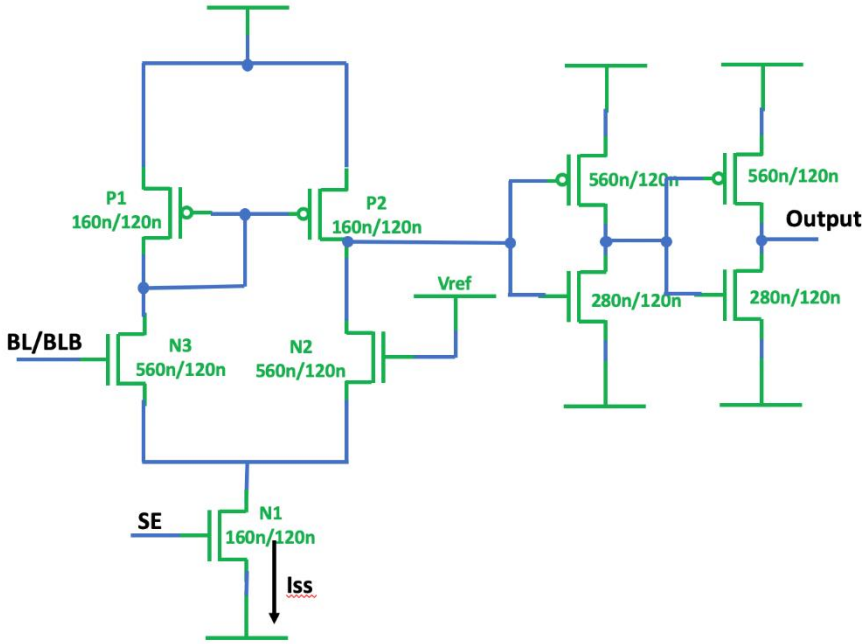
SRAM cell

- Discharge stability: N>P, P2/P3>P4
- Leakage control: N1/N2↓



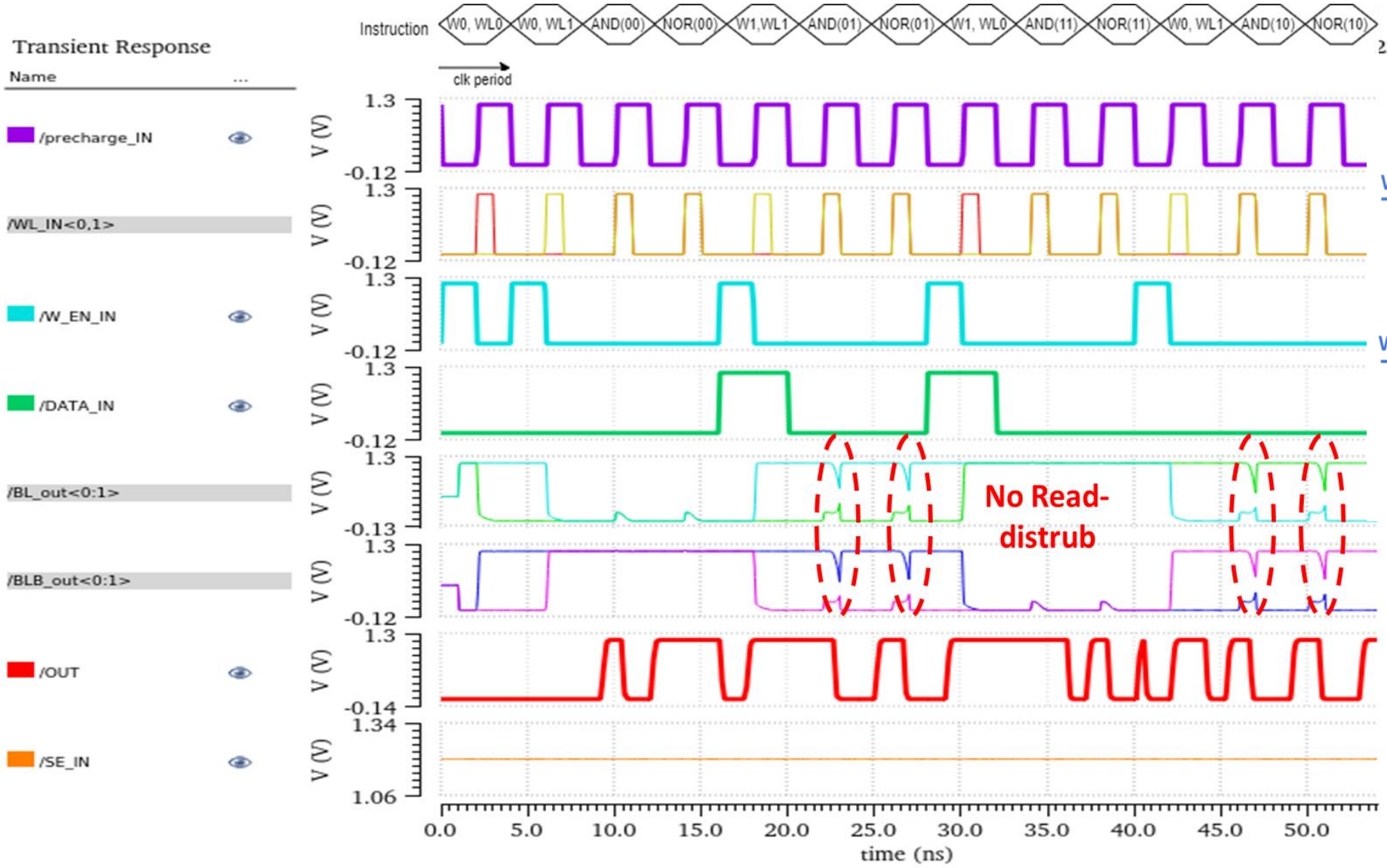
Precharge & discharge

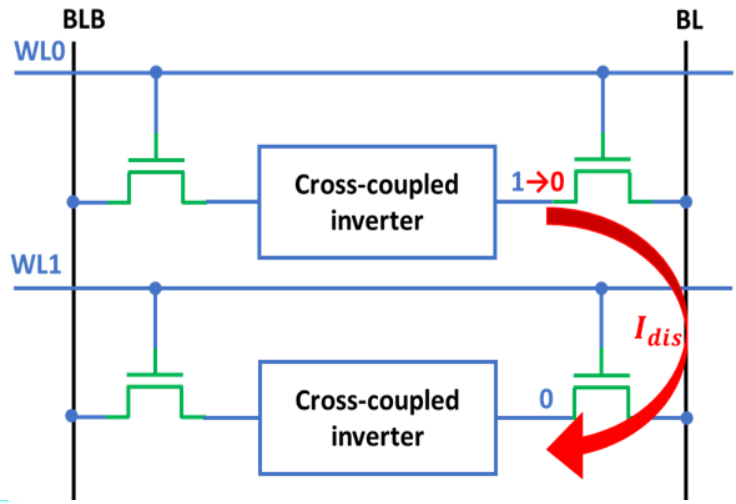
- Current mirror: P1=P2
- Power control: N1 ↓
- High to low: N2>P2



Sense amplifier

Data Disturbing

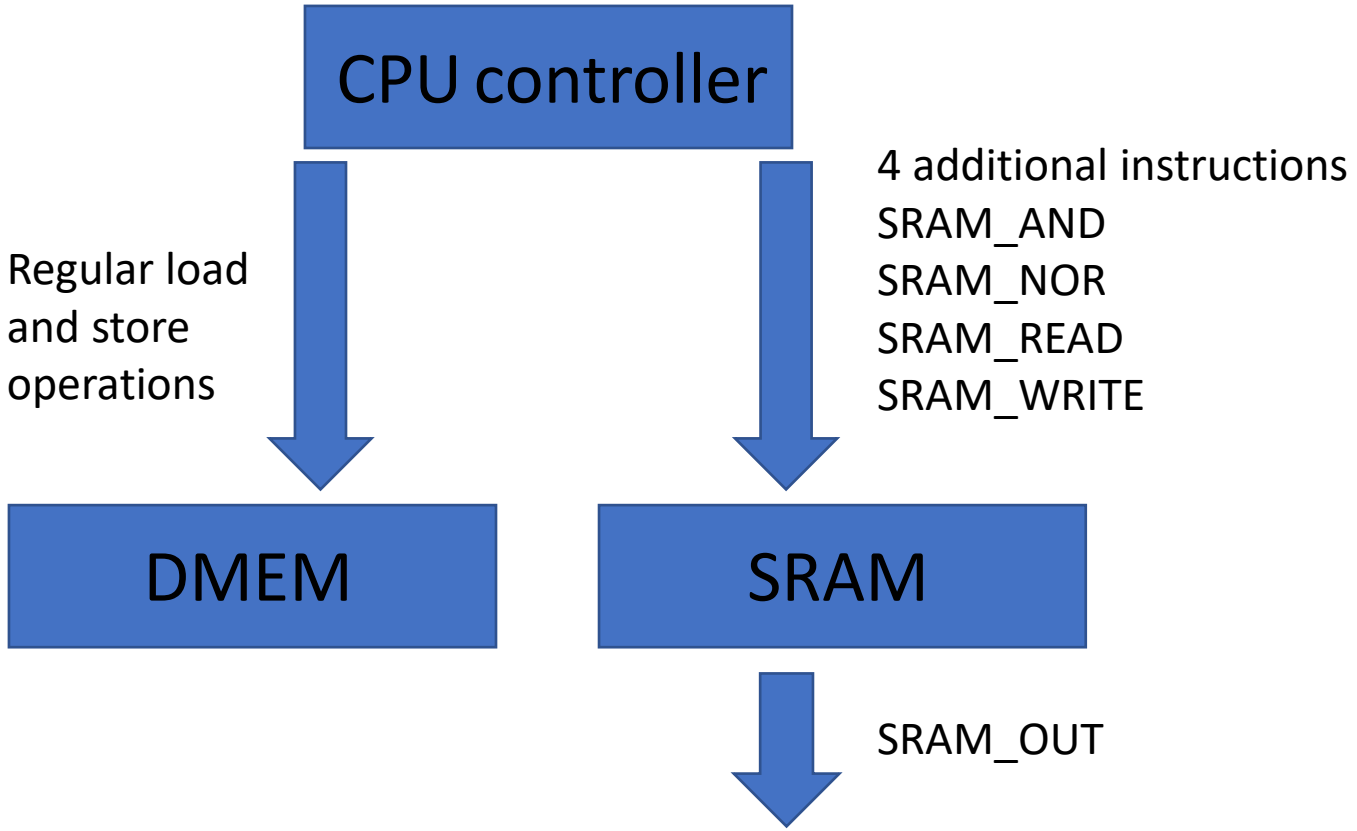




Solution: reduce WL activation duty cycle

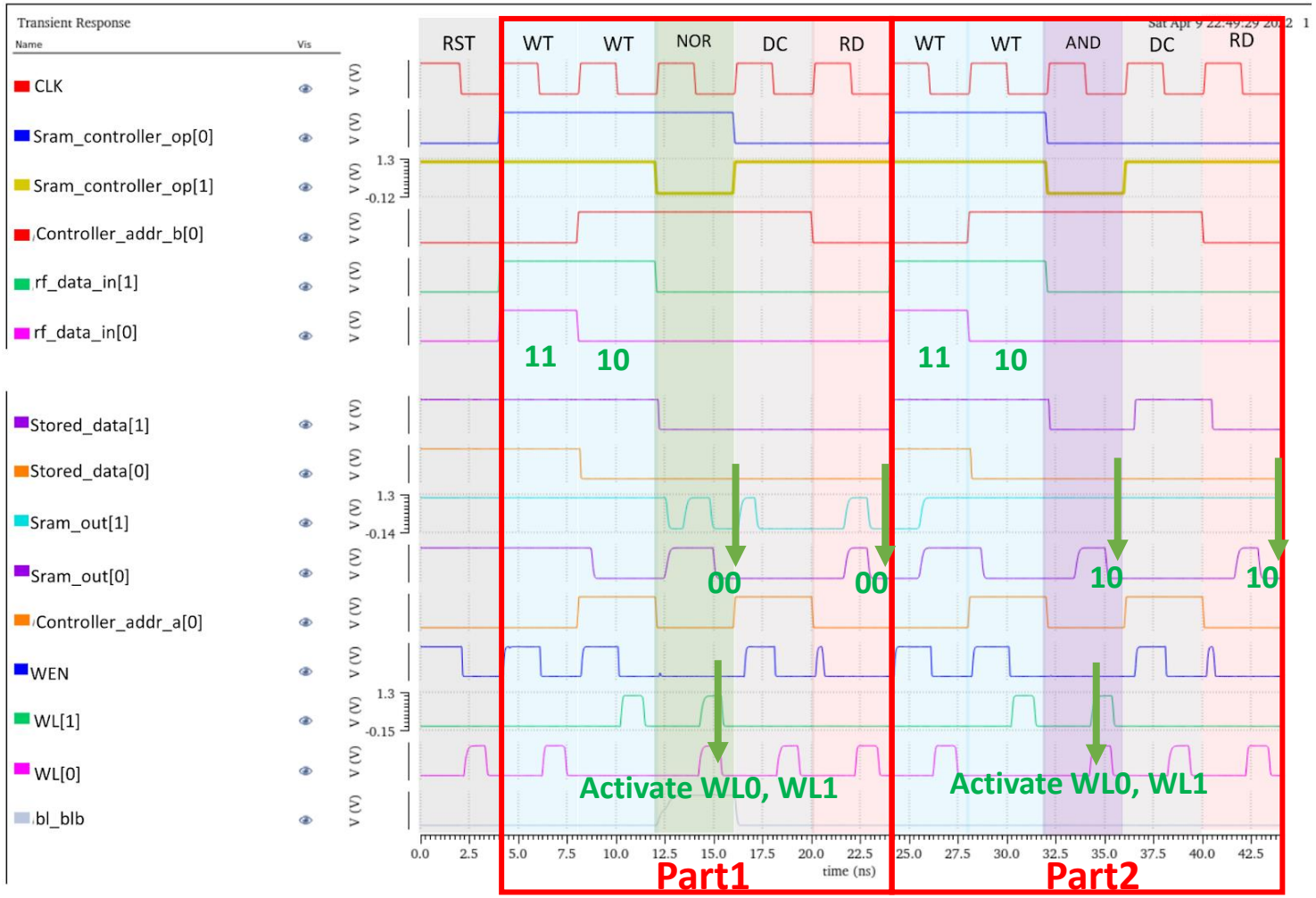
Integration with Baseline Processor

- SRAM and DMEM are two memory module with the same hierarchical level.
- Four additional instructions from CPU controller to SRAM



SRAM SPICE Simulation

- Testcase
  - Part1:NOR
  - Part2:AND
- Examining the read result in the end of the read cycle

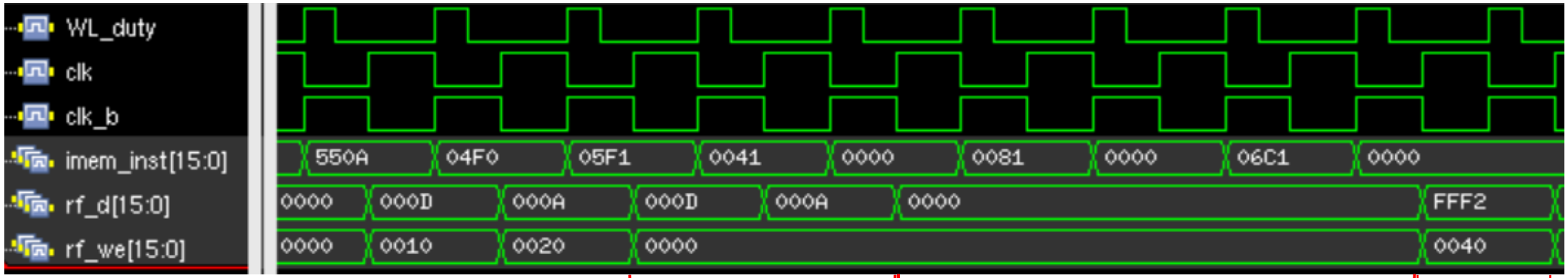


Simulation of SRAM with Baseline Processor

- IMEM not available in SPICE simulation…
  - Simulate using NC-Verilog
  - Build a behavioral model for SRAM cells in Verilog
- Waveform

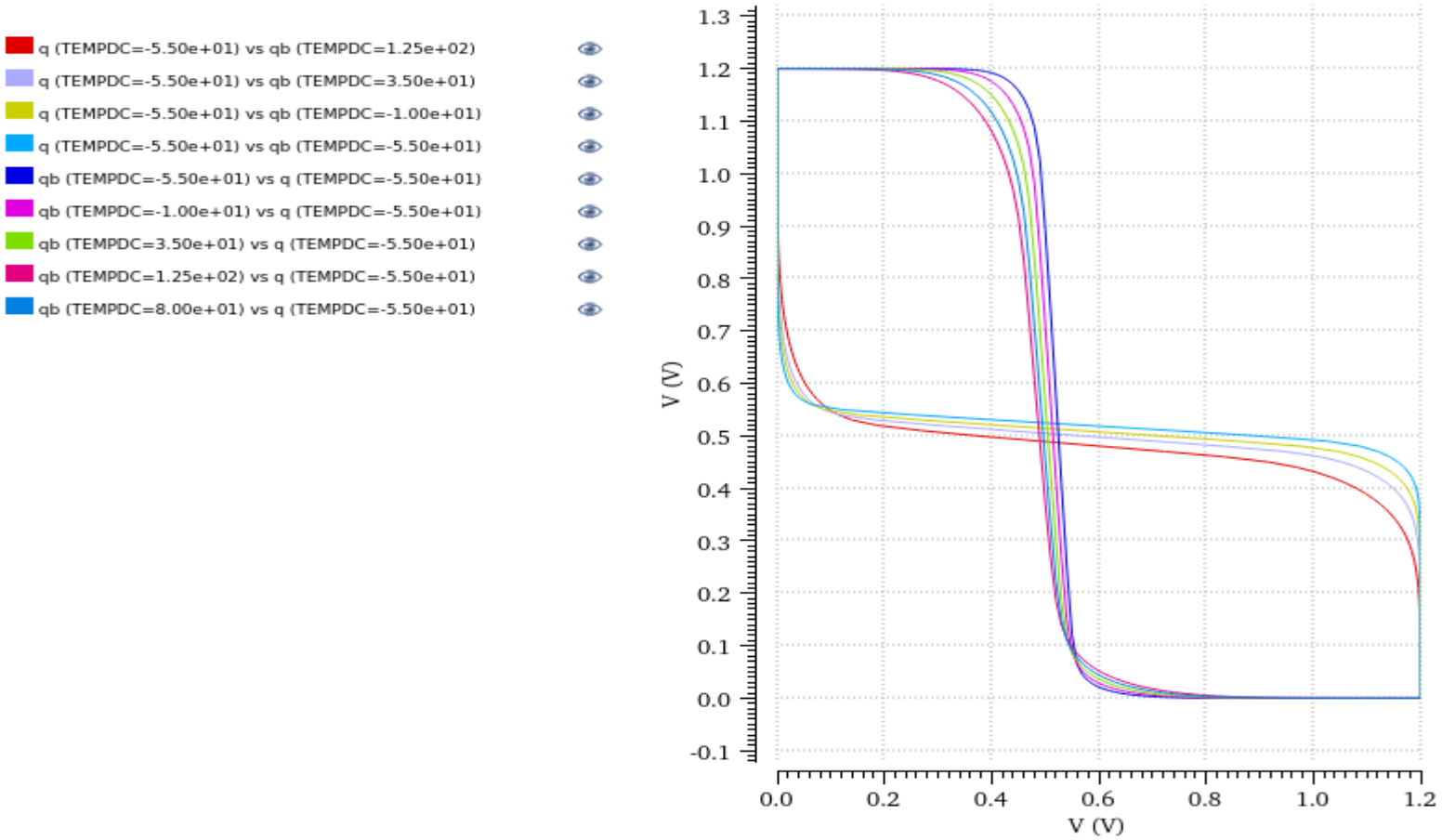
- Testing assembly code:

Instruction	Operation	Note	
540D	reg4=D		
550A	reg5=A		
04F0	SRAM[0]=reg4		store to SRAM
05F1	SRAM[1]=reg5		
0041	SRAM[1]=SRAM[0] & SRAM[1]	0x000D & 0x000A = 0x0008	
0000	nop		in-SRAM boolean operation
0081	SRAM[1]=SRAM[0] ~| SRAM[1]	0x000D ~| 0x0008 = 0xFFF2	
0000	nop		
06C1	reg6=SRAM[1]		load from SRAM



Static Noise Margin Analysis

- Sweep the temperature from -55℃ to 125 ℃



Conclusion

What we learn:

- The efficeincy of baseline processor can be improved by implementing the SRAM in memory computation
- The SRAM cell needs to be carefully sized for Read & Write operations
- Data disturbing issue is solved by adjusting the duty cycle of the WL

Progress:

- Finished: SRAM layout and functionality test
- Ongoing: layout integration with baseline processor and SRAM
- Future work: Monte Carlo simulation